It is claimed:

1. A method of controlling independent and asynchronous access to a memory by a plurality of processes, the method comprising:

while providing for independent and asynchronous performance of a memory read process of the plurality of processes, independently and asynchronously performing a memory write process of the plurality of processes, wherein the memory write process comprises:

obtaining a first copy of a memory full indicator;

obtaining a copy of a read index, the read index copy indicating a read element position in the memory;

when the first copy of the memory full indicator indicates that the memory is not full, determining an available write count from the read index copy and a write index, the write index indicating a write element position in the memory;

beginning at the write element position, writing an amount of data corresponding to the available write count; and

updating the write index to indicate a next write element position based upon the amount of data written.

2. The method of claim 1, further comprising:

determining whether the updated write index is equal to the read index copy.

3. The method of claim 2, further comprising:

when the updated write index is equal to the read index copy, setting the memory full indicator to indicate that the memory is full or may be full.

4. The method of claim 1, wherein the amount of data corresponding to the available write count is an amount of data which is less than or equal to the available write count.

5. The method of claim 1, wherein the memory read process comprises:

obtaining a second copy of the memory full indicator;

obtaining a copy of the write index;

determining an available read count from the read index and the write index copy;

beginning at the read element position, reading an amount of data corresponding to the available read count; and

updating the read index to indicate a next read element position based on the amount of data read.

6. The method of claim 5, further comprising:

subsequent to updating the read index, when the second copy of the memory full indicator indicates that the memory is full or may be full, clearing the memory full indicator to indicate that the memory is not full.

7. The method of claim 5, further comprising:

prior to determining the available read count, when the second copy of the memory full indicator indicates that the memory is full or may be full, and when the write index copy is not equal to the read index, clearing the memory full indicator to indicate that the memory is not full.

8. The method of claim 5, wherein the amount of data corresponding to the available read count is an amount of data which is less than or equal to the available read count.

9. The method of claim 1, wherein the memory read process comprises:

obtaining a second copy of the memory full indicator;

obtaining a copy of the write index;

determining an available read count from the read index and the write index copy;

beginning at the read element position, reading a plurality of data elements corresponding to the available read count;

selecting a data element of the plurality of data elements for a subsequent read process;

determining a corresponding element position of the data element in the memory as a next read element position; and

updating the read index to indicate the next read element position for the subsequent read process.

10. The method of claim 1, wherein the memory read process further comprises:

obtaining a second copy of the memory full indicator;

obtaining a copy of the write index;

determining an available read count from the read index and the write index copy;

beginning at the read element position, reading a plurality of data elements corresponding to the available write count;

selecting a subset of data elements from the plurality of data elements for a subsequent read process;

determining an element position corresponding to an initial data element of the subset of data elements in the memory, as a next read element position and as a next write element position;

commencing at the next write element position, writing the subset of data elements to the memory; and

updating the read index to indicate the next read element position for the subsequent read process.

11. The method of claim 10, wherein the writing of the subset of data elements to the memory is performed without modification of the write index.

12. A system for controlling independent and asynchronous access to a memory by a plurality of processes, the system comprising:

a memory;

a first node coupled to the memory, the first node capable of independently and asynchronously performing a memory read process of the plurality of processes; and

a second node coupled to the memory, the second node capable of independently and asynchronously performing a memory write process, of the plurality of processes, by obtaining a first copy of a memory full indicator and obtaining a copy of a read index, the read index copy indicating a read element position in the memory; when the first copy of the memory full indicator indicates that the memory is not full, the second node further capable of determining an available write count from the read index copy and a write index, the write index indicating a write element position in the memory; beginning at the write element position, the second node further capable of writing an amount of data corresponding to the available write count and updating the write index to indicate a next write element position based upon the amount of data written.

13. The system of claim 12, wherein the second node is further capable of determining whether the updated write index is equal to the read index copy.

14. The system of claim 13, wherein when the updated write index is equal to the read index copy, the second node is further capable of setting the memory full indicator to indicate that the memory is full or may be full.

15. The system of claim 12, wherein the amount of data corresponding to the available write count is an amount of data which is less than or equal to the available write count.

16. The system of claim 12, wherein the first node is further capable of obtaining a second copy of the memory full indicator and obtaining a copy of the write index; determining an available read count from the read index and the write index copy; beginning at the read element position, reading an amount of data corresponding to the available read count; and wherein the first node is further capable of updating the read index to indicate a next read element position based on the amount of data read.

17. The system of claim 16, wherein subsequent to updating the read index, when the second copy of the memory full indicator indicates that the memory is full or may be full, the first node is further capable of clearing the memory full indicator to indicate that the memory is not full.

18. The system of claim 16, wherein prior to determining the available read count, when the second copy of the memory full indicator indicates that the memory is full or may be full, and when the write index copy is not equal to the read index, the first node is further capable of clearing the memory full indicator to indicate that the memory is not full.

19. The system of claim 16, wherein the amount of data corresponding to the available read count is an amount of data which is less than or equal to the available read count.

20. The system of claim 12, wherein the first node is capable of obtaining a second copy of the memory full indicator and obtaining a copy of the write index; determining an available read count from the read index and the write index copy; beginning at the read element position, reading a plurality of data elements corresponding to the available read count; and wherein the first node is further capable of selecting a data element of the plurality of data elements for a subsequent read process; determining a corresponding element position of the data element in the memory as a next read element position; and updating the read index to indicate the next read element position for the subsequent read process.

21. The system of claim 12, wherein the first node is capable of obtaining a second copy of the memory full indicator and obtaining a copy of the write index; determining an available read count from the read index and the write index copy; beginning at the read element position, reading a plurality of data elements corresponding to the available write count; selecting a subset of data elements from the plurality of data elements for a subsequent read process; wherein the first node is further capable of determining an element position corresponding to an initial data element of the subset of data elements in the memory, as a next read element position and as a next write element position; commencing at the next write element position, writing the subset of data elements to the memory; and updating the read index to indicate the next read element position for the subsequent read process.

22. The system of claim 21, wherein the first node is further capable of writing the subset of data elements to the memory without modification of the write index.

23. The system of claim 12, wherein the first node and the second node are selected from a plurality of nodes, the plurality of nodes comprising one or more of the following node types: an adaptive computing node, a kernel node, a processor, and a finite state machine.

24. A method of controlling independent and asynchronous access to a memory by a plurality of processes, the method comprising:

while providing for independent and asynchronous performance of a memory write process of the plurality of processes, independently and asynchronously performing a memory read process of the plurality of processes, wherein the memory read process comprises:

obtaining a first copy of a memory full indicator;

obtaining a copy of a write index, the write index indicating a write element position in the memory;

determining an available read count from the write index copy and a read index, the read index indicating a read element position in the memory;

beginning at the read element position, reading a plurality of data elements corresponding to the available read count; and

updating the read index to indicate a next read element position.

25. The method of claim 24, wherein the updating of the read index to indicate the next read element position is based upon the plurality of data elements read.

26. The method of claim 24, further comprising:

subsequent to updating the read index, when the first copy of the memory full indicator indicates that the memory is full or may be full, clearing the memory full indicator to indicate that the memory is not full.

27. The method of claim 24, further comprising:

prior to determining the available read count, when the first copy of the memory full indicator indicates that the memory is full or may be full, and when the write index copy is not equal to the read index, clearing the memory full indicator to indicate that the memory is not full.

28. The method of claim 24, wherein the plurality of data elements corresponding to the available read count is an amount of data which is less than or equal to the available read count.

29. The method of claim 24, further comprising:

prior to updating the read index, selecting a data element from the plurality of data elements for a subsequent read process; and

determining a corresponding element position of the data element in the memory as the next read element position.

30. The method of claim 24, further comprising:

prior to updating the read index, selecting a subset of data elements from the plurality of data elements for a subsequent read process;

determining an element position corresponding to an initial data element of the subset of data elements in the memory, as the next read element position and as a next write element position; and

commencing at the next write element position, writing the subset of data elements to the memory.

31. The method of claim 30, wherein the writing of the subset of data elements to the memory is performed without modification of the write index.

32. The method of claim 24, wherein the memory write process further comprises:

obtaining a second copy of the memory full indicator;

obtaining a copy of the read index;

when the second copy of the memory full indicator indicates that the memory is not full, determining an available write count from the read index copy and the write index;

beginning at the write element position indicated by the write index, writing an amount of data corresponding to the available write count; and

updating the write index to indicate a next write element position based upon the amount of data written.

33. The method of claim 32, further comprising:

determining whether the updated write index is equal to the read index copy.

34. The method of claim 33, further comprising:

when the updated write index is equal to the read index copy, setting the memory full indicator to indicate that the memory is full or may be full.

35. The method of claim 32, wherein the amount of data corresponding to the available write count is an amount of data which is less than or equal to the available write count.

36. A system for controlling independent and asynchronous access to a memory by a plurality of processes, the system comprising:

a memory;

a first node coupled to the memory, the first node capable of independently and asynchronously performing a memory write process of the plurality of processes; and

a second node coupled to the memory, the second node capable of independently and asynchronously performing a memory read process, of the plurality of processes, by obtaining a first copy of a memory full indicator and obtaining a copy of a write index, the write index indicating a write element position in the memory; determining an available read count from the write index copy and a read index, the read index indicating a read element position in the memory; beginning at the read element position, reading a plurality of data elements corresponding to the available read count and updating the read index to indicate a next read element position.

37. The system of claim 36, wherein the second node is further capable of updating the read index to indicate the next read element position based upon the plurality of data elements read.

38. The system of claim 36, wherein subsequent to updating the read index, the second node is further capable, when the first copy of the memory full indicator indicates that the memory is full or may be full, of clearing the memory full indicator to indicate that the memory is not full.

39. The system of claim 36, wherein prior to determining the available read count, the second node is further capable, when the first copy of the memory full indicator indicates that the memory is full or may be full and when the write index copy is not equal to the read index, of clearing the memory full indicator to indicate that the memory is not full.

40. The system of claim 36, wherein the plurality of data elements corresponding to the available read count is an amount of data which is less than or equal to the available read count.

41. The system of claim 36, wherein prior to updating the read index, the second node is further capable of selecting a data element from the plurality of data elements for a subsequent read process, and determining a corresponding element position of the data element in the memory as the next read element position.

42. The system of claim 36, wherein prior to updating the read index, the second node is further capable of selecting a subset of data elements from the plurality of data elements for a subsequent read process; determining an element position corresponding to an initial data element of the subset of data elements in the memory, as the next read element position and as a next write element position; and commencing at the next write element position, writing the subset of data elements to the memory.

43. The system of claim 42, wherein the second node is further capable of writing the subset of data elements to the memory without modifying the write index.

44. The system of claim 36, wherein the first node is further capable of obtaining a second copy of the memory full indicator and obtaining a copy of the read index; when the second copy of the memory full indicator indicates that the memory is not full, determining an available write count from the read index copy and the write index; beginning at the write element position indicated by the write index, writing an amount of data corresponding to the available write count; and updating the write index to indicate a next write element position based upon the amount of data written.

45. The system of claim 44, wherein the first node is further capable of determining whether the updated write index is equal to the read index copy.

46. The system of claim 45, wherein the first node is further capable, when the updated write index is equal to the read index copy, of setting the memory full indicator to indicate that the memory is full or may be full.

47. The system of claim 44, wherein the amount of data corresponding to the available write count is an amount of data which is less than or equal to the available write count.

48. The system of claim 36, wherein the first node and the second node are selected from a plurality of nodes, the plurality of nodes comprising one or more of the following node types: an adaptive computing node, a kernel node, a processor, and a finite state machine.

49. A system for controlling independent and asynchronous access to a memory by a plurality of processes, the system comprising:

a memory;

a first node coupled to the memory, the first node independently and asynchronously capable of obtaining a first copy of a memory full indicator and obtaining a copy of a write index, the write index indicating a write element position in the memory; determining an available read count from the write index copy and a read index, the read index indicating a read element position in the memory; beginning at the read element position, reading a plurality of data elements corresponding to the available read count; the first node further capable of selecting a subset of data elements from the plurality of data elements for a subsequent read process and when selected, determining an element position corresponding to an initial data element of the subset of data elements in the memory, as a next read element position and as a next write element position, and commencing at the next write element position, writing the subset of data elements to the memory; and the first node further capable of updating the read index to indicate the next read element position and, when the first copy of the memory full indicator indicates that the memory is full or may be full, clearing the memory full indicator to indicate that the memory is not full; and

a second node coupled to the memory, the second node independently and asynchronously capable of obtaining a second copy of the memory full indicator and obtaining a copy of the read index; when the second copy of the memory full indicator copy indicates that the memory is not full, determining an available write count from the read index copy and the write index; beginning at the write element position indicated by the write index, writing an amount of data corresponding to the available write count; updating the write index to indicate a next write element position based upon the amount of data written; and when the updated write index is equal to the read index copy, of setting the memory full indicator to indicate that the memory is full or may be full.